WHAT IS CLAIMED IS:

1. A method for providing contextual diagnostic data at a point of failure of a software program, comprising:

(a) registering callbacks for one or more modules and sub-applications within the program;

(b) examining a call stack for the program upon failure of the program;

(c) notifying the registered callbacks for the modules and sub-applications based on the examined call stack;

(d) performing callback processing, wherein the notified callbacks of the modules and sub-applications extract and supply context data; and

(e) packaging the context data supplied by the notified callbacks of the modules and sub-applications.

2. The method of claim 1, wherein the registering step (a) comprises registering callbacks for the modules and sub-applications when an address of a procedure or function within the modules and sub-applications is on the call stack upon the failure of the program.

3. The method of claim 1, wherein the context data is comprised of stack data, heap data, global data or external data.

4. The method of claim 1, wherein the packaging step (e) comprises storing the packaged context data.

5. The method of claim 1, wherein the packaging step (e) comprises transferring the packaged context data to a server computer.

6. The method of claim 5, wherein the packaging step (e) comprises storing the transferred, packaged context data on the server computer.

7. An apparatus for providing contextual diagnostic data at a point of failure of a software program, comprising:

a computer; and

logic, performed by the computer, for:

(a) registering callbacks for one or more modules and sub-applications within the program;

(b) examining a call stack for the program upon failure of the program;

(c) notifying the registered callbacks for the modules and sub-applications based on the examined call stack;

(d) performing callback processing, wherein the notified callbacks of the modules and sub-applications extract and supply context data; and

(e) packaging the context data supplied by the notified callbacks of the modules and sub-applications.

8. The apparatus of claim 7, wherein the logic for registering (a) comprises logic for registering callbacks for the modules and sub-applications when an address of a procedure or function within the modules and sub-applications is on the call stack upon the failure of the program.

9. The apparatus of claim 7, wherein the context data is comprised of stack data, heap data, global data or external data.

10. The apparatus of claim 7, wherein the logic for packaging (e) comprises logic for storing the packaged context data.

11. The apparatus of claim 7, wherein the logic for packaging (e) comprises logic for transferring the packaged context data to a server computer.

12. The apparatus of claim 11, wherein the logic for packaging (e) comprises logic for storing the transferred, packaged context data on the server computer.

13. An article of manufacture embodying logic for providing contextual diagnostic data at a point of failure of a software program, the logic comprising:

(a) registering callbacks for one or more modules and sub-applications within the program;

(b) examining a call stack for the program upon failure of the program;

(c) notifying the registered callbacks for the modules and sub-applications based on the examined call stack;

(d) performing callback processing, wherein the notified callbacks of the modules and sub-applications extract and supply context data; and

(e) packaging the context data supplied by the notified callbacks of the modules and sub-applications.

14. The article of claim 13, wherein the registering step (a) comprises registering callbacks for the modules and sub-applications when an address of a procedure or function within the modules and sub-applications is on the call stack upon the failure of the program.

15. The article of claim 13, wherein the context data is comprised of stack data, heap data, global data or external data.

16. The article of claim 13, wherein the packaging step (e) comprises storing the packaged context data.

17. The article of claim 13, wherein the packaging step (e) comprises transferring the packaged context data to a server computer.

18. The article of claim 17, wherein the packaging step (e) comprises storing the transferred, packaged context data on the server computer.